**USER GUIDE**

**Weapon Detection Using Faster R-CNN**

**🧾 1. Introduction**

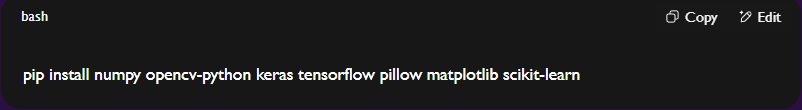
This guide explains how to install, set up, and operate the Weapon Detection System developed using Python, TensorFlow, OpenCV, and Tkinter. The system can detect weapons like guns and knives from both images and videos using a GUI-based interface.

**🧰 2. Prerequisites**

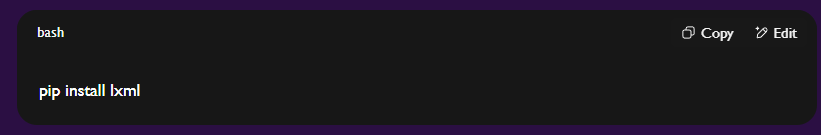
✅ Software Requirements

| Component | Version / Tool |
| --- | --- |
| OS | Windows 10 or above |
| Python | 3.7.0 |
| IDE (recommended) | Visual Studio Code / PyCharm |

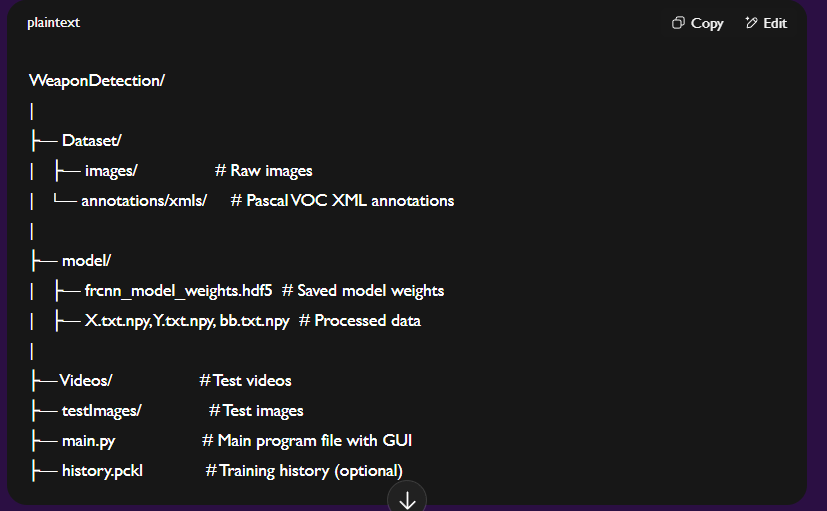
✅ Python Libraries (Install using pip)



You may also need:



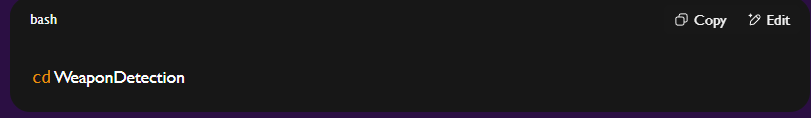
**🗂️ 3. Folder Structure**

****

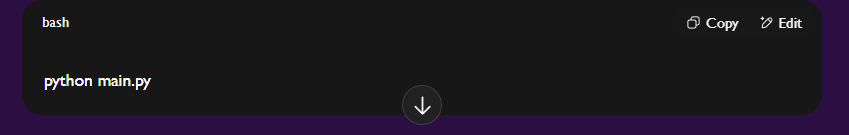
**🖥️ 4. How to Run the Program**

**Step 1: Launch the Application**

* Open a terminal/command prompt.
* Navigate to the project directory:



* Run the GUI:



**Step 2: GUI Overview**

Once launched, you will see a radar-style GUI with buttons:

* Upload Weapon Dataset
* Generate & Load Weapon Detection Model
* Upload Image
* Detect Weapon from Image
* Detect Weapon from Video
* FRCNN Training Accuracy-Loss Graph

🕹️ **5. Step-by-Step Instructions**

🟩 **5.1 Upload Dataset**

* Click "Upload Weapon Dataset"
* Choose the folder that contains annotations/xmls and images/
* The system will either:
  + Load preprocessed .npy files if available
  + Or process the XML files and generate them

🟩 **5.2 Train & Load Model**

* Click "Generate & Load Weapon Detection Model"
* If a trained model exists (frcnn\_model\_weights.hdf5), it will load directly.
* Otherwise, the model will train using the loaded dataset.

📝 The console log will display:

* Model accuracy, precision, recall, and F1 score

🟩 **5.3 Upload Image & Detect**

* Click "Upload Image" and select an image from testImages/
* Click "Detect Weapon from Image"
* The model will process the image and show bounding boxes if weapons are found.

🟩 **5.4 Detect from Video**

* Click "Detect Weapon from Video" and select a video file from Videos/
* A video window opens, showing detection frame by frame.
* Press q to exit video detection.

🟩 5**.5 View Accuracy-Loss Graph**

* Click "FRCNN Weapon Detection Training Accuracy-Loss Graph"
* Displays accuracy and loss graphs using matplotlib

**🗒️ 6. Notes & Tips**

* Only supported formats: .jpg, .png, .mp4, .avi
* Make sure model/ directory exists and is writable
* For accurate detection, input images should be clear and annotated properly
* You may re-train the model by deleting the frcnn\_model\_weights.hdf5 file

**❌ 7. Troubleshooting**

| **Issue** | **Solution** |
| --- | --- |
| "ModuleNotFoundError" | Install missing libraries using pip install |
| "Image not loading" | Ensure correct path and file format |
| "Model not detecting" | Recheck dataset or retrain model |
| "GUI crashes" | Run from terminal to check error messages |

**📤 8. Exporting and Logs**

* Logs are displayed in the GUI console (Text widget)
* You can copy/save them manually
* Detected image/video results can be saved using OpenCV’s imwrite() (add in code if needed)

**🔁 9. Maintenance and Future Use**

* Keep model and data files backed up
* Update datasets and retrain periodically
* Extend for live webcam/CCTV input or Raspberry Pi deployment (Future Scope)